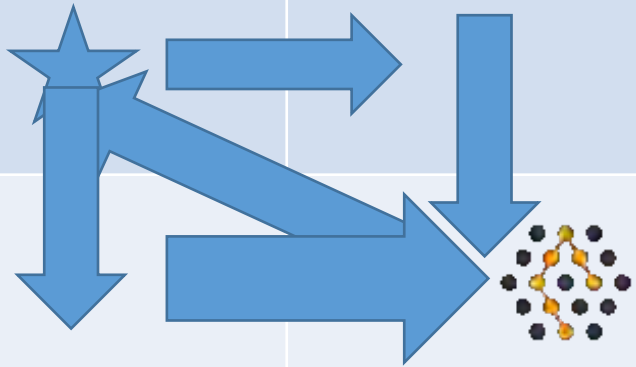
改变我人生的软件交付质量

宫载军@明略数据
gongzaijun@mininglamp.com

项目管理的围城

	外企	创业团队
技术		
管理		

职业生涯的两次奇点

从技术到管理

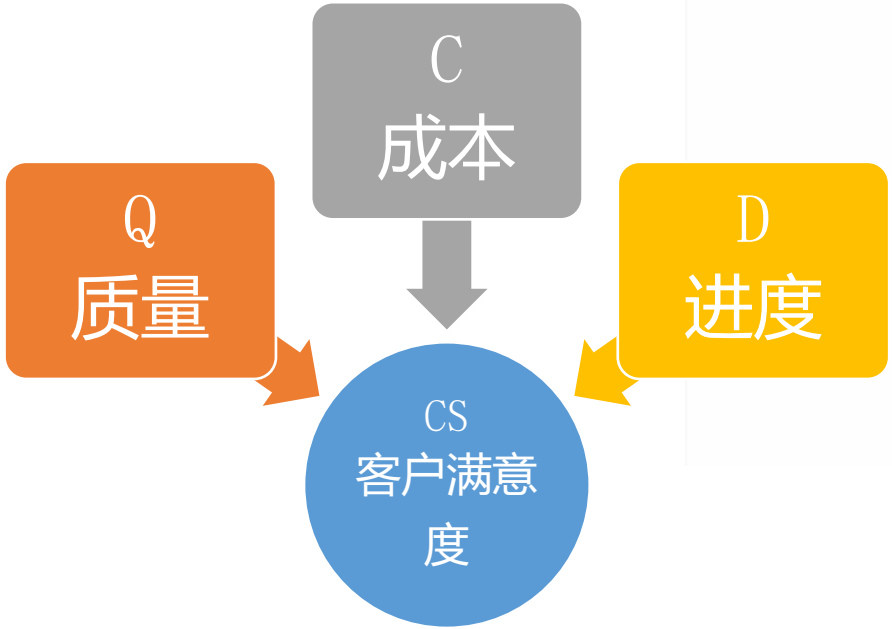
从小团队管理到大规模集团军管理

“这两年当PM，把自己都做废了”

- 如果只打杂，不关心技术、不关心业务、不关心客户、不关心团队、不积极发现项目问题、不拼命去找解决办法，自然把自己做废了

- 项目管理的方法可以应用在各个领域，包括你的生活、你的家庭追女朋友、管理家庭、儿子上学…… 都是一个Project

项目管理的3个维度



一切从第一个交付开始改变

- 没有一份报纸价格算错
- 开始联调后没事情做了

工作进入正循环

高质量代码

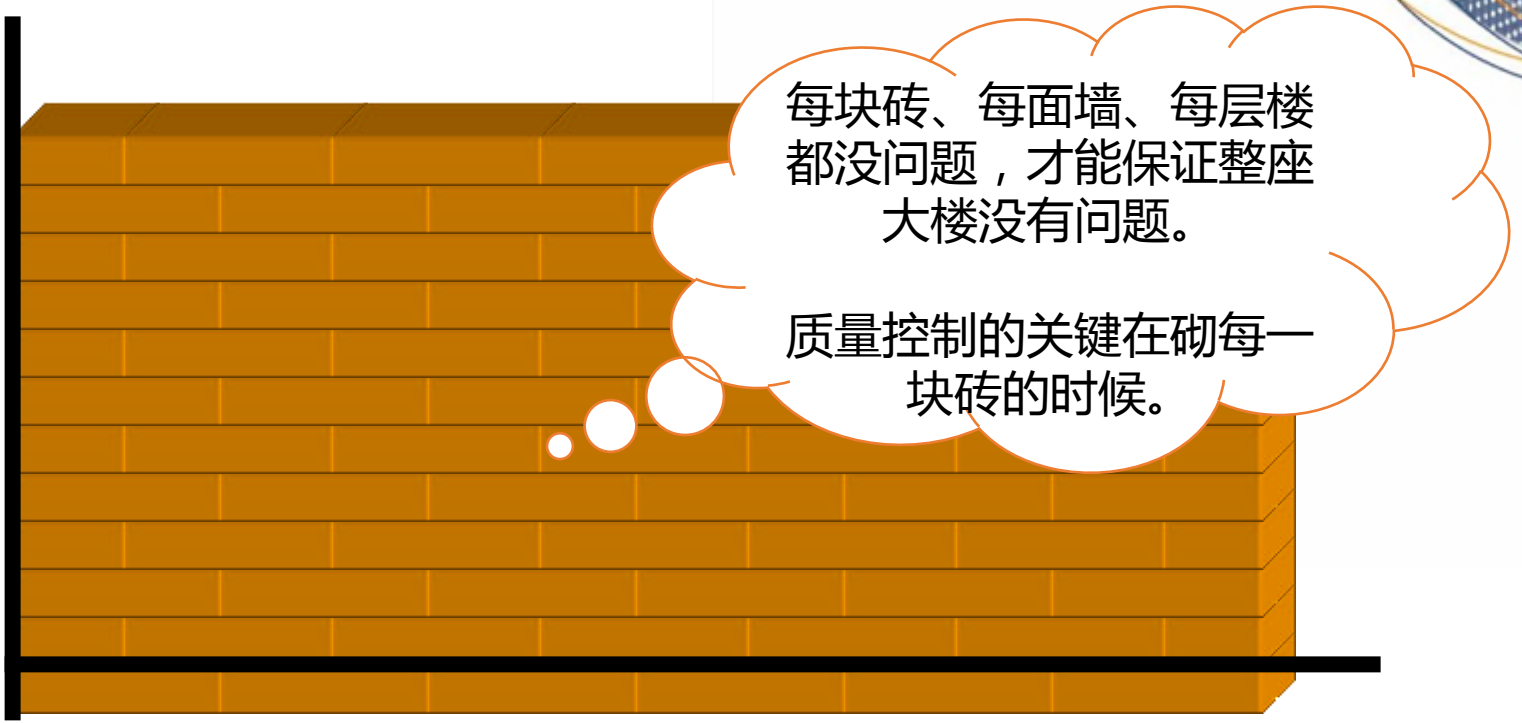
- 被认同、享受工作乐趣
- 别人支持信任
- 拥抱新项目、新技术

差质量代码

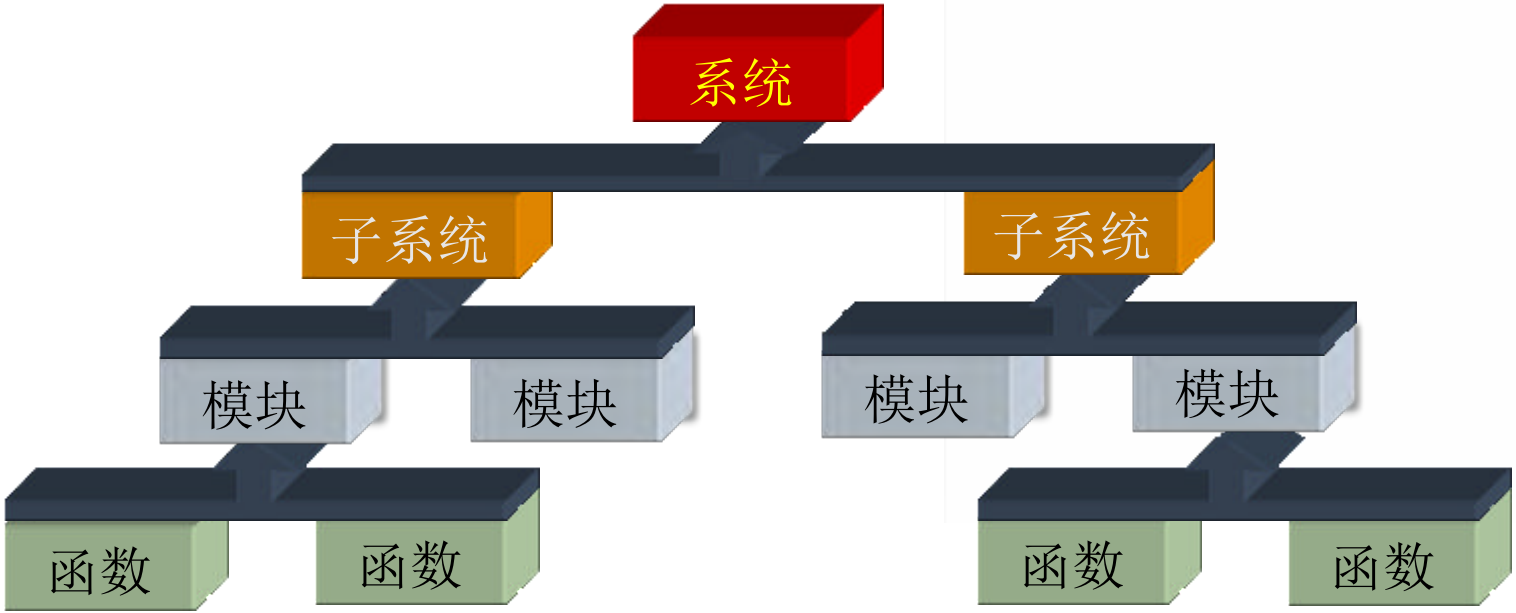
- 无休止的BugFix
- 应对质疑
- 累、烦、失落

怎么保证代码质量？

两根绳子：建筑工程质量控制方法



软件开发系统的构成



不能把程序质量拜托给测试人员

是咱们没把楼盖好，没资格责备检测人员没发现问题

- ✓ 楼房盖歪了再验收，补再多水泥也不会是好房子
- ✓ 同样，代码写的时候不注意质量，测试发现问题再修改也不会是好代码

第一部分：

程序员，如何管理代码质量

写程序第一原则：想清楚再写

- 细节想不清楚的，可以先写框架
- 在写代码的过程中理清楚所有逻辑
- 最终提交的代码，必须是所有情况都想清楚了的

✓ 能力高低，就在于你考虑得全不全
✓ 越拼命去想，能力提升越快

考虑所有的情况

没有“应该不会发生”的情况

- 访问文件/数据库有可能访问不了
- 文件格式/表结构有可能和约定的不一致
- 有可能取空值/多值
- 除0
- 越界
- 调别人函数/别人调你函数，都有可能没按约定来实现；
  今天说好不处理这种情况，今后可能会变；
- ……

像安保一样，进门先搜身

界面输入、入口参数、数据库/文件中取出来的值、调其它函数返回来的值等等，所有从外面进来的东西，先判断一下是否有问题。

进门先搜身，就算来捣乱也只能扔扔鞋。

典型的代码风格问题

```
if( 正常条件 ) {
    正常逻辑处理;
} else {    // 异常条件
    异常处理;
    return;
}
```

```
If( 异常条件 ) {
    异常处理;
    return;
}

正常逻辑处理;
```

写程序第二原则：能追溯

- Bug是肯定存在的
- 写代码的时候就要考虑Bug发生时，如何在最短时间发现问题所在

让程序自己能Debug

TRACE运行结果

- 有问题第一时间捕获

记录必要的信息

- 留下脚印：记录程序运行的踪迹
- 留下证据：出错时的各种状况

提示信息要清晰准确

- 以后来解决问题的不一定是你自己
- 即使是自己也可能会忘记的
- 提示清楚了客户可能自己就能解决

让代码像书本一样易读


```
Function1()
{
      Initialize();

      Step1();
      Step2();
      Step3();

       Finally()
}
```

```
Step1.1();
Step1.2();
```

```
Step1.1.1();
Step1.1.2();
```

明略数据 MININGLAMP

自我检出Bug的2个法宝

- 桌上Debug
- 单元测试

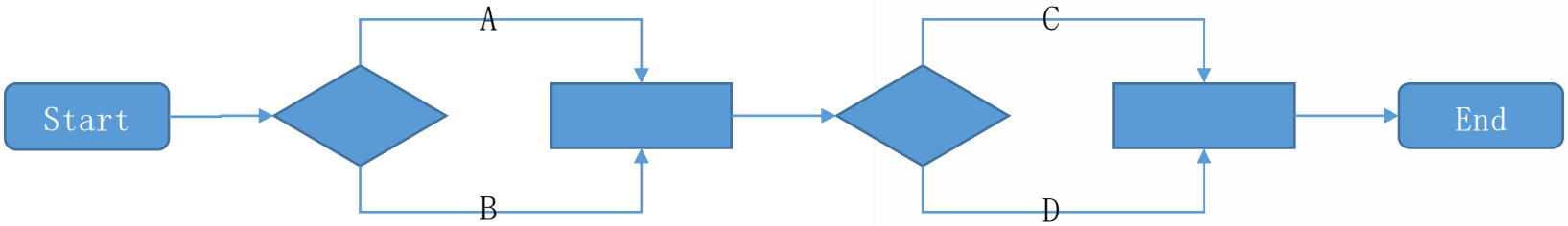
桌上Debug

写完程序自己先读一遍

- 都是按照自己设计的思路实现的吗？
- 回头再看看是否有考虑漏的情况吗？

单元测试

- 遍历分支
  不可能遍历所有的条件组合，但可能发生的分支务必要确认到。



- CD/桌上Debug后尽早做UT、UT一定要自己做！
  UT也应该是对代码最熟悉的时候、最熟悉的人做。

匠心：自己给自己的信念

与具体的技巧相比，拥有一颗程序员的心才是最重要的。

- 不是因为公司有规定，我才要检查每个分支。
- 不要因为有人Review，就认为有错也没关系。

每一行代码都是我们自己的作品，我们有责任让他成为最好。

一个项目有很多自己无法控制的地方，总会出很多问题，但对一名程序员而言，唯一的信条应该是“不要让问题出在我的代码上！”

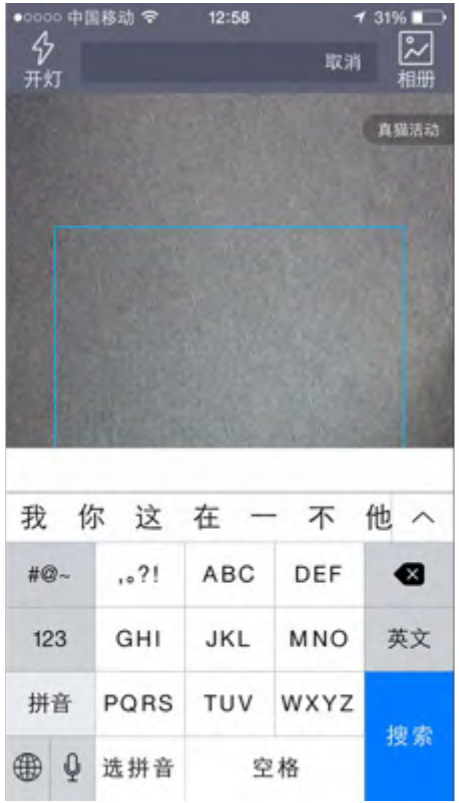
匠心：程序=作品=精品

- 只有想着做精品，才有可能做精品
- 只有持续出精品，才能成长飞跃
- 只有出精品，才能被认同，获得更多机会

明略最被认同的团队

明略数据
MININGLAMP

小细节，用心去关注



明略数据 MININGLAMP

第二部分

团队，如何管理代码质量

质量，不是靠流程管理

- 是技术能力的一部分
- 是融入在团队血液中

让每个人都认同，包括客户！

团队保证质量的方法

Review

高手辅导

新手学习

第三方测试

更多场景确认

平台级问题发现

质量标准

允许瑕疵	拒绝缺陷

和一栋大厦一样：
一个系统的好坏是由每一行代码这些一块一块砖头的质量决定的

就像盖了座危楼一样：
提交没有质量保证的代码，其危害比没提交代码更严重

不能通过加班加资源盖危楼！

- 工作量评估要考虑Review、UT的时间
- 资源不够，应减少Feature，优先保证重要的，提交有质量保证的代码

- ✓ 可以提供半成品，不能提供次品
- ✓ 提交没质量保证的代码，就像盖了危楼，危害比没有盖楼更严重
- ✓ 不能通过加班加资源盖出更多危楼！

不贰过 – 多问几个为什么

美国有个护士给病人发错了药

- 中国式处理方法（不问为什么）
  这个护士是临时工，已被开除

- 美国式处理方法（问为什么）
  - ✓ 看病历记录，病人增加30%，而护士人手没有增加
    → 对策：增加人手
  - ✓ 看药，外形和颜色极相似，容易混淆
    → 对策：给药厂发函，区分药品外包装或形状

PM=教练

挑人：合适
训练：严格
比赛：审时度势
总结：持续改进

明略数据
MININGLAMP